

41.6 LUA Grundlagen - Ein erstes Skript

Autor:Goetz

Quelle: Mein EEP-Forum

Die vorherigen Kapitel reichen noch nicht, um irgendetwas auf einer Anlage per Lua zu steuern.

Aber es wird Zeit einmal auszuprobieren, ob man Lua nun überhaupt dazu bringen kann etwas zu tun. Und zwar nicht irgendwas. Sondern exakt das, was man will.

Erinnert euch an eure erste Fahrstunde (falls ihr nicht zu jung seid und die erst noch vor euch habt? Da seid ihr nicht gleich ins Nachbardorf gefahren. Und auch nicht zum Bäcker um die Ecke. Ihr habt überhaupt nichts gemacht, was irgendwie mit dem Verkehr auf unseren Straßen zu tun hatte. Sondern ihr habt Kreise auf einem großen Parkplatz gedreht, um zunächst ein Gefühl für das Auto zu bekommen.

Und ich weiß noch heute, wie überrascht ich dabei war. Das ging nämlich alles nicht so leicht, wie ich erwartet hatte.

Anfahren, Lenken, Bremsen ...

Was von außen sooo leicht ausgesehen hatte, war plötzlich ziemlich schwer.

Und es erforderte mehr Konzentration als ich geahnt hatte.

Mit dem Auto namens "Lua" ist das nicht anders.

Dafür muss man erst einmal ein Gefühl bekommen. Das kann man nämlich auch "abwürgen".

Deshalb möchte ich vorschlagen, dass ihr das mit einer neuen, komplett leeren Anlage versucht. Die ist sozusagen euer großer Parkplatz für die ersten Versuche.

Sobald die leere Anlage geöffnet ist, macht ihr das Skript Fenster auf und löscht alles, was dort drin steht. Komplett!

Also auch die **EEPMain()** Funktion.

Damit - um beim Vergleich mit dem Parkplatz zu bleiben - nichts im Weg steht, was ihr anrempeln könntet.

Und dann schreibt ihr in das Skript Fenster nur eine einzige Zeile:

```
print("Lua rennt")
```

Das wird sozusagen euer erster Blick in den Rückspiegel.

Wir müssen den Rückspiegel jetzt nur noch finden ...

Sucht unten am Fensterrand den Knopf "Skript neu laden" und drückt ihn.

Das ist nötig, damit EEP das übernimmt, was ihr in das Skript Fenster geschrieben habt.

Das Skript Fenster schließt sich und das Skript, welches ihr eben geschrieben habt, wird an EEP übertragen.

Öffnet die Programmeinstellungen und sucht in der rechten Spalte ganz unten "EEP Ereignis Fenster" Das muss aktiviert sein, wenn man die Ausgabe von **print()** sehen möchte.

Und wenn ihr jetzt in den 3D Modus der Anlage wechselt, dann müsstet ihr im Ausgabefenster die Schrift

Lua rennt sehen.

Also den Text, den ihr zwischen die Klammern hinter **print** geschrieben habt. Aber ohne die Anführungszeichen.

Das sieht nicht so spannend aus wie ein 20-gleisiger, vollautomatisierter Schattenbahnhof mit Zufallssteuerung.

Aber weil so wenig passiert, kann man ganz genau hinschauen und jedes Detail bemerken.

Und genau darum geht es.

Also zum Beispiel um die Tatsache, dass die Anführungsstriche nicht mit ausgegeben werden. Weil sie ein Steuerzeichen sind welches Lua benötigt, um Buchstaben als Text zu erkennen und nicht als Variablen.

Außerdem steht noch etwas im Ausgabefenster. Nämlich
Error running function 'EEPMain': attempt to call a nil value

Merkwürdig, oder?

Wir haben doch gar keine EEPMain Funktion mehr im Skript. Wie kann die da eine Fehlermeldung produzieren?

Zeit für einen weiteren Test, um der Sache auf die Spur zu kommen.

Dafür baut ihr jetzt bitte in das Skript eine eigene Funktion ein.

Öffnet also das Skriptfenster.

Dort sollte jetzt

```
print("Lua rennt")
```

zu lesen sein. Genau, wie ihr es rein geschrieben hattet.

Lasst dahinter (für bessere Lesbarkeit) eine Zeile frei und schreibt dann

```
function druckMich()
```

```
print("Hier meldet sich die Funktion")
```

```
end
```

Mehr nicht!

Drückt wieder auf "Skript neu laden" unten rechts im Skriptfenster.

Sofort wird das Fenster geschlossen und das Skript übertragen.

Außerdem wird es sofort ausgeführt, wenn ihr im 3D Fenster geblieben seid.

Falls ihr vorher in den 2D Modus gewechselt habt, dann müsst ihr jetzt wieder zurück zu 3D, damit das neue Skript ausgeführt wird.

Im Ausgabefenster sollte jetzt stehen

Lua rennt

Error running function 'EEPMain': attempt to call a nil value

Lua rennt

Error running function 'EEPMain': attempt to call a nil value

Nicht ärgern, dass das nicht das ist, was ihr erwartet hattet. Sondern Schlüsse daraus ziehen.

Das geht besser, wenn ihr gleich noch eine Änderung am Skript vornehmt.

Öffnet das Skriptfenster wieder und ändert die erste Zeile in

```
print("Lua verwirrt mich")
```

aber lasst den Rest so, wie er ist.

Drückt "Skript neu laden" und im Ausgabefenster müsste jetzt stehen:

```
Lua rennt
Error running function 'EEPMain': attempt to call a nil value
Lua rennt
Error running function 'EEPMain': attempt to call a nil value
Lua verwirrt mich
Error running function 'EEPMain': attempt to call a nil value
```

Aaaahhh ... Das Ausgabefenster wird also nicht automatisch beim Start gelöscht. Die ersten vier Zeilen sind von den ersten beiden Versuchen stehen geblieben.

Okay, das ist schonmal eine nützliche Erkenntnis.

Und wo ist der Text aus dem **print()** Befehl, den wir in die Funktion **druckMich()** geschrieben haben?

Nicht zu sehen ...

Weil diese Funktion nie aufgerufen wird. Der Befehl **function** sorgt nur dafür, dass Lua lernt, was in dieser Funktion zu tun ist.

Nächster Test:

Öffnet bitte wieder das Skript. Geht ganz ans Ende und fügt eine weitere Zeile ein:

```
druckMich()
```

Das ganze Skript sollte nun so aussehen:

```
print("Lua verwirrt mich")

function druckMich()
print("Hier meldet sich die Funktion")
end

druckMich()
```

Anschließend "Skript neu laden" und im Ausgabefenster sollte jetzt folgendes stehen:

```
Lua rennt
Error running function 'EEPMain': attempt to call a nil value
Lua rennt
Error running function 'EEPMain': attempt to call a nil value
Lua verwirrt mich
Error running function 'EEPMain': attempt to call a nil value
Lua verwirrt mich
Hier meldet sich die Funktion
Error running function 'EEPMain': attempt to call a nil value
```

Die letzten drei Zeilen sind bei diesem Test hinzu gekommen. Der Rest ist das, was wir bei den vorigen Versuchen ausgegeben hatten.

Okay, eine Funktion muss also aufgerufen werden, damit sie überhaupt etwas tun kann.

Zeit zu probieren was passiert, wenn man eine Funktion aufruft, die es nicht gibt.

Erweitert euer Skript am Ende um eine weitere Zeile:

```
michGibtsNicht()
```

und achtet unbedingt auf die Klammern am Ende. Die sagen Lua, dass hier eine Funktion aufgerufen werden

soll.

Lasst das Skript laufen und ihr bekommt:

Lua rennt

Error running function 'EEPMain': attempt to call a nil value

Lua rennt

Error running function 'EEPMain': attempt to call a nil value

Lua verwirrt mich

Error running function 'EEPMain': attempt to call a nil value

Lua verwirrt mich

Hier meldet sich die Funktion

Error running function 'EEPMain': attempt to call a nil value

Lua verwirrt mich

Hier meldet sich die Funktion

Error running function 'michGibtsNicht': attempt to call a nil value

Error running function 'EEPMain': attempt to call a nil value

Bemerkenswert ist hier die vorletzte Zeile!

Wenn man eine Funktion aufruft, die nicht definiert wurde, dann kommt genau die Fehlermeldung, die wir schon beim ersten Mal (für die Funktion EEPMain) gesehen hatten.

Ihr wisst, wo die Funktion **michGibtsNicht()** aufgerufen wird. Ihr habt den Aufruf ja selbst ins Skript geschrieben.

Aber wer ruft denn dauernd diese Funktion **EEPMain()** auf, die es gar nicht gibt? Die steht doch nirgendwo im Skript?

Die Antwort ist: EEP macht das. Dieser Aufruf ist fest in EEP eingebaut.